The first computer program is generally dated to 1843, when mathematician Ada Lovelace published an algorithm to calculate a sequence of Bernoulli numbers, intended to be carried out by Charles Babbage's Analytical Engine..  
 Machine code was the language of early programs, written in the instruction set of the particular machine, often in binary notation.  
 Various visual programming languages have also been developed with the intent to resolve readability concerns by adopting non-traditional approaches to code structure and display.  
For example, when a bug in a compiler can make it crash when parsing some large source file, a simplification of the test case that results in only few lines from the original source file can be sufficient to reproduce the same crash.  
 Allen Downey, in his book How To Think Like A Computer Scientist, writes:  
 Many computer languages provide a mechanism to call functions provided by shared libraries.  
Many applications use a mix of several languages in their construction and use.  
 Whatever the approach to development may be, the final program must satisfy some fundamental properties.  
 Different programming languages support different styles of programming (called programming paradigms).  
Ideally, the programming language best suited for the task at hand will be selected.  
However, because an assembly language is little more than a different notation for a machine language, two machines with different instruction sets also have different assembly languages.  
 Code-breaking algorithms have also existed for centuries.  
There exist a lot of different approaches for each of those tasks.  
Unreadable code often leads to bugs, inefficiencies, and duplicated code.  
Text editors were also developed that allowed changes and corrections to be made much more easily than with punched cards.  
However, Charles Babbage had already written his first program for the Analytical Engine in 1837.